Cloud-Based Intelligent Surveillance for Digital Forensics: AI-Enhanced Criminal Investigations

Aseel Abdullah Aljuhani, Fatima Hamed Aljuhani
Department of Cybersecurity, College of Computer and Information Sciences,
Northern Border University (NBU), Arar 73222, Saudi Arabia

***Abstract*—Modern smart surveillance systems have become a core element of digital forensics workflows, offering real-time detection of weapons, fire, smoke, blood, cars, individuals, and other related objects. These systems improve evidence collection, accelerate threat identification, and enhance investigative efficiency. However, most traditional surveillance architectures operate without sufficient digital forensic awareness, evidence integrity mechanisms, standardized evidence management, or reliable methods for detecting tampering. These systems rarely support chain of custody documentation and often lose reliability under adverse conditions. To address these shortcomings, a unified system that aggregates all suspicious objects and movements is needed, geared towards forensics. Therefore, we developed VigilEye, a cloud-oriented forensic surveillance framework that combines public surveillance datasets with simulated crime scene scenarios. It applies digital forensic preprocessing, such as CLAHE, to improve clarity, and SHA-256 hashing and metadata to ensure evidence integrity. Previous studies have also highlighted privacy challenges; we addressed this by sending blurred images for alerts, while the original, unblurred images are designed to be securely stored in an encrypted evidence environment accessible only to authorized personnel. This study presents the design, implementation, and digital forensic evaluation of VigilEye. The model achieved promising experimental performance (mAP50 ≈ 0.72) with an average detection speed of 33.55 ms per image, indicating that further optimization is required to enhance both speed and accuracy. The system sends alerts via the Telegram platform if it detects suspicious behavior or wanted individuals. VigilEye demonstrates the feasibility of a forensic-aware surveillance workflow with real-time alerting and integrity-preserving evidence handling in an experimental setting. Our future work includes expanding digital forensic datasets, improving spatial-temporal event detection, automating cloud-based chain of custody management, and enhancing interoperability with tools such as Autopsy and FTK Imager, alongside strengthening privacy safeguards and connected camera digital forensic tracking.**

***Keywords*—VigilEye; digital forensics; YOLOv8; real-time object detection; evidence integrity**

I. Introduction

Sophisticated and intelligent surveillance systems are critical components of today's digital forensic process. Over the past several years, these systems have enabled the collection and real-time analysis of visual indicators of potential evidence [1]. Visual indicators include weapons, smoke, fire, blood, vehicles, and human presence. The sheer number and type of visual indicators available allow investigators a significant opportunity to create actionable knowledge from the moment an event occurs. The use of these systems increases both the speed at which an investigator can respond to incidents and the maintenance of situational awareness throughout the entire event [2]. As a consequence, conventional surveillance has transitioned from a primarily passive recording function to a dynamic digital forensic mechanism in support of criminal investigations, emergency response efforts and the criminal justice system itself [3], [4]. While improvements have been made in the use of artificial intelligence and vision systems to detect situations or behaviors that might constitute criminal activity, the vast majority of traditional surveillance systems are standalone modules that typically operate independently from one another to perform a single unique function.

Often, although not exclusively, these systems will be used in conjunction with a digital forensic investigation; however, they will lack a number of the basic information safeguards necessary in order to produce evidence of digital forensic quality, including evidence of authentication and traceability [1], [5]. Their use is therefore not routinely accepted as admissible in a court of law. Very few traditional surveillance systems follow standardized procedures for evidence management, correctly complete chain of custody documentation, or implement any means of verifying the integrity of their evidence through the use of tamper checks or other mechanisms [5]. As a result, while the accuracy of the detection model of a camera system might be high, the digital forensic evidence from that camera system may not satisfy the requirements of a digital forensic investigation, which will limit the utility and reliability of digital evidence that might otherwise have significant consequences in the courtroom [1], [5].

Finally, the existing models, datasets and pipeline systems for detecting objects or events within video files are poorly designed to remain stable under realistic environmental conditions [6], [2]. A variety of factors, including poor lighting conditions, substandard or broken camera equipment, and a range of environmental challenges, can negatively affect the quality of digital forensic videos as well as adversely affect detection accuracy [6]. These same factors lead to higher incidence of false-positive and false-negative digital forensic detection results, both of which undermine the clarity and confidence needed for digital forensic evaluation and presentation in court [7], [2]. In addition, privacy preservation has become an essential requirement for forensic awareness surveillance systems [8]. Because of these challenges, there is a growing demand for an integrated surveillance system that detects all suspicious objects and movements, is designed for digital forensics purposes, and maintains privacy.

To address this gap, the present study introduces VigilEye, an AI-based forensic surveillance system integrating advanced object detection techniques with forensic preprocessing, evidence integrity safeguards and addressing privacy issues. VigilEye uses publicly available surveillance footage combined with simulated crime examples to create realistic learning conditions. YOLOv8 and YOLOv5 detection models, along with preprocessing techniques such as CLAHE, and SHA-256 hashing, are used to detect high-risk forensic indicators.

II. Related Work

Artificial intelligence and surveillance camera systems have recently contributed significantly to criminal investigations; therefore, researchers have attempted to combine them to enhance their effectiveness in supporting both digital and physical forensics.

Shah et al. [9] focused on an algorithm to detect weapons in public places. The model achieved an average response time of 1.30 seconds, although the model requires numerous resources and a large storage space. Dutta et al. [10] proposed a model to enhance violence detection [3D CNN + Bi-LSTM technology] and achieved higher efficiency and accuracy; however, it faced difficulty in performing in crowded places and low-lighting. Bashambu et al. [11] developed a real-time fire and smoke detection system using YOLOv5, which can accurately identify fire and smoke in CCTV footage and handle it quickly with little delay. However, because the system relies only on visual input, its accuracy may decrease in low-light environments or when fire and smoke are too small or combine with similar-looking objects. Khalil et al. [12] created a real-time blood detection system for CCTV using an attention-enhanced InceptionV3 model. The system showed high accuracy in difficult scenes and different lighting conditions. However, it needs large, labeled datasets and strong computing power, which can limit its use in real-time, especially in places with limited resources.

Hamdi et al. [13] proposed a technique to improve license plate identification accuracy from 30.59% to 74.78%; however, the model required more data and training time compared to other algorithms. In addition, Vats et al. [14] introduced an accident detection system that uses YOLOv3 to detect vehicle frames and Vision Transformers to identify accidents in real time. Although this method improves accuracy and reduces processing costs, it may still have issues in complex situations, such as when vehicles are very close to each other or are already damaged. These cases can cause the system to misclassify accidents.

Arnab et al. [15] showed that transformers outperformed 3D CNNs, achieving advanced accuracy and good performance; however, for training on small datasets, regularization techniques must be applied. Xiao et al. [1] proposed using the CLAHE algorithm to improve video quality, and the accuracy improved significantly after using the algorithm.

Sholahuddin et al. [16] attempted to optimize YOLOv8 and showed that the model achieved high speed and acceptable accuracy in the field; however, they discovered a slight decrease in accuracy, but low-light and video compression may lead to false and inaccurate details. Wang et al. [17] proposed the Eagle YOLOv8 model to enhance object detection. The model increased accuracy by 8.56%, mAP50 by 10.06%, and recall by 9.43%. However, further improvements in performance and deployment are still required. Potter et al. [6] focused on proposing a system to improve night vision in surveillance cameras. The system attained about 71.5% accuracy at 20 frames per second during testing; however, the system requires active learning to correct errors and improve accuracy.

Ayyan Zubair [3] introduced the DAS system in New York City explaining how it uses surveillance cameras to collect and store video that can be accessed in real-time by police and private organizations. Although the system is useful, it still presents privacy risks. To address privacy problems, M. Kassir, S. Haidar, and A. Yaacoub [18] introduced a solution using personalized federated learning (PFL). The model reached 99.3% accuracy in detecting aggressive acts while still protecting privacy. However, the system still faces challenges when processing complex video environments in real-time settings.

Pisati et al. [7] conducted a comparison of algorithms to clarify the strengths and weaknesses of different models [CNN, RNN, hybrid models]. Sonawane et al. [2] compared various algorithms used to detect suspicious activities. Naheed Akhtar et al. [5] demonstrated methods for detecting video manipulation and concluded that modern technologies help enhance the reliability of digital evidence. Diksha et al. [19] reviewed facial recognition techniques and explained how open-source tools such as Dlib, face-recognition, and OpenCV can support real-time surveillance systems.

Mahdi et al. In [20], an experiment was carried out to train a model that could identify suspect faces and notify security personnel when a match was detected. The system showed good performance; however, it struggled in difficult conditions such as low-light, varied facial angles, and long-range distances. Toshpulatov et al. [21] examined the challenge of protecting individual privacy in CCTV footage and proposed a practical method for face anonymization prior to analysis. Their method uses recent face detection models like RetinaFace and TinaFace, together with simple but effective OpenCV operations. The results suggest that using accurate face detection with OpenCV blurring helps protect privacy in surveillance footage, especially when showing someone's identity may cause ethical or legal issues.

Although numerous studies have investigated intelligent surveillance cameras to support investigations, each has focused on a separate aspect of intelligent surveillance. To the best of our knowledge, no academic research has addressed an intelligent surveillance system capable of recognizing faces, plates, blood, suspicious behavior, and car accidents while simultaneously providing real-time alerts within a unified cloud-oriented framework to support digital forensic investigations. This study aims to address this gap.

In VigilEye, we selected YOLOv8-based detection techniques because they provide fast and reliable real-time performance, which is essential for surveillance applications. We also used simple image enhancement methods, for instance CLAHE, to improve visibility in low-light scenarios without increasing model complexity. Alternatively, we did not rely on heavy or highly specialized models, such as standalone

transformer architectures or task-specific networks, because they were not necessary for training our model or achieving real-time detection within the VigilEye framework.

Unlike previous studies that focus on isolated detection tasks, VigilEye integrates multi-class detection, forensic preservation, and real-time alerting within a unified cloud-oriented framework.

The main contributions of this work are summarized as follows:

- A unified cloud-oriented forensic surveillance framework (VigilEye).
- Integration of real-time AI detection with digital forensic evidence preservation.
- Implementation of SHA-256 hashing and metadata tracking to ensure evidence integrity and traceability.
- A privacy-preserving alerting mechanism using face anonymization techniques.
- A real-time alert system integrated with Telegram to enable rapid incident response.

III. Methodology

A. Research Design for Digital Forensics

This study follows an applied experimental research design. The design guarantees that forensic evidence is preserved through all stages of evidence processing, including dataset preparation, preprocessing, model training, inference, metadata generation, hashing and storage, and real-time alerts. In contrast with conventional studies that assess discrete tasks (e.g., face identification, vehicle registration analysis), VigilEye integrates these components into one unified forensic workflow. This includes fire monitoring, smoke detection, weapon identification, human detection, blood spotting, cars, vehicle accident identification, license plate assessment, and fight detection. VigilEye's cloud-oriented architecture allows investigators to use a single pipeline for digital forensics to collect, analyze, and store all forms of digital evidence and associated metadata.

B. System Architecture with Forensic Integration

As shown in Fig. 1, the VigilEye system is designed to ensure immediate detection while preserving evidence. It integrates real-time monitoring, secure preprocessing, evidence encryption, and alert mechanisms into a unified workflow.

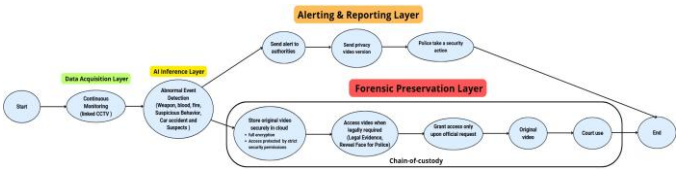

Fig. 1. VigilEye system

The system operates through four core layers:

1) Data acquisition layer (camera and input sources): Live video is imported from surveillance cameras or uploaded media. Frames are sampled at specific time intervals to balance speed and forensic completeness. This includes optional image enhancement using CLAHE when visibility is poor, while preserving the original frames for evidentiary purposes.

2) AI Inference layer (detection models): Each frame is run through a multi-model inference engine consisting of YOLOv8-based detectors for most object detection categories and a YOLOv5-based component for person detection. This modular design allows the models to be implemented together and also allows each class to operate independently with its own confidence thresholds and optimization settings.

IV. Implementation and Forensic Preservation

A. Forensic Preservation Layer

To support the forensic reliability and traceability of digital evidence, the system generates an SHA-256 hash for each original frame before any enhancement. It also creates a preservation data string containing timestamps, model version information, and processing steps. Duplicate copies of both original and enhanced images are retained. Furthermore, immutable logging mechanisms are employed to ensure traceability and non-repudiation. The system aligns with SWGDE [22] and NIST SP 800-86 [23] guidelines for digital evidence integrity.

B. Alerting and Reporting Layer

Instant alerts are sent via the Telegram app, containing the image, classification, confidence level, hash, and metadata. The alert system is designed to be lightweight and fast, ensuring immediate situational awareness while preserving the quality of forensic evidence.

The system's architecture showcases the strengths of real-time performance achieved through optimized GPU models, high-quality digital forensics processing via hashing and metadata chaining, a flexible modular design that facilitates model addition and replacement, and support for digital forensics reconstruction, including chronological analysis and reidentification. Each feature contributes a unique forensic signature. Every piece of evidence detected is recorded in a secure, tamperproof forensics log containing a timestamp, confidence scores, and each alert links investigators to the secure forensics repository.

C. Data Collection and Forensic Preprocessing

The data included general surveillance footage and simulated crime scene scenarios with variations in lighting, obscuring, motion blurring, and environmental distortion. Dataset sizes were fire and smoke: 400 images [24], weapons: 442 images [25], people: 690 images [26], blood: 160 images [27], car accidents: 987 images [28], license plate: 300 images [29], fight: 802 images [30], cars: 800 images [31], and faces: 1000 images [32]. The preprocessing included CLAHE low-light correction technology. This technology resulted in significant improvements in visibility, particularly with regard to fire, smoke, and vehicle license plates.

D. Privacy-Preserving Preprocessing

A privacy protection module was used to automatically blur faces in all output frames, in order to maintain ethical and legal compliance. The anonymized version was used for live

monitoring and alerts, while the unblurred original was securely encrypted and stored in the forensic evidence vault. Hashes (SHA-256) were generated for both versions to verify integrity and ensure a complete chain of custody. This separation allows investigators to use privacy-preserving outputs for routine monitoring while retaining the original evidence only for authorized forensic review. Therefore, the system balances operational awareness with evidence protection and privacy requirements.

E. Model Development with Forensic Objectives

The model development prioritized real-time detection and forensic reliability. YOLOv8 was adopted as the primary architecture for most object-detection categories within the VigilEye system due to its strong performance and stability. YOLOv5 was used for the person-detection component, while the remaining object-detection categories were implemented using YOLOv8. Face recognition was handled separately as a matching component rather than a general object-detection task. Key methodological decisions included stronger lowlight preprocessing for fire and smoke detection, augmentation for the blood dataset due to its small size, balanced samples for accident detection, and class isolation improvements for weapons detection. This clarification improves reproducibility by distinguishing the final reported model configuration from earlier exploratory trials conducted during system development.

F. Cloud-Oriented Deployment Design for Forensic Investigations

The VigilEye system was designed as a cloud-oriented forensic surveillance framework that supports evidence acquisition, preprocessing, object detection, SHA-256 integrity verification, secure storage, and automatic alerting. In the proposed operational deployment, the evidence vault would be implemented using encrypted cloud storage with access control, audit logging, and secure transmission mechanisms. However, in the current experimental implementation, model training and evaluation were conducted using Google Colab and Kaggle due to resource limitations, and the generated evidence artifacts were stored locally within the Kaggle environment. Therefore, the term cloud-based in this study refers to the intended system architecture and cloud-ready workflow rather than a fully production-deployed cloud infrastructure. This distinction clarifies that the present work evaluates the feasibility of a forensic-aware surveillance pipeline, while full cloud deployment, multi-user access control, and long-term evidence lifecycle management remain part of future implementation work.

G. Implementation Tools and Forensic Environment

As shown in Fig. 2, during early experiments, Google Colab was used for training; however, the free tier provided very limited GPU time and frequent session interruptions, which made full-scale training impractical. For this reason, the training pipeline was migrated to Kaggle, where the free environment offers longer, more stable GPU sessions, enabling consistent model training and evaluation. The VigilEye system was implemented using Ultralytics YOLOv8 and YOLOv5 for object detection, OpenCV (cv2) for image processing, face blurring, and frame extraction, NumPy for numerical operations, the face-recognition library for embedding-based face matching, and hashlib (SHA-256) for cryptographic hashing. Although the proposed architecture supports Python-based cloud storage, the current experimental prototype stored evidence of artifacts locally within the Kaggle environment. This decision was due to the resource and access limitations of the available free-tier platforms. Alerts were sent via a developed Telegram Bot API, where successful requests returned HTTP 200 responses, confirming message delivery. This implementation demonstrates the functional feasibility of the alerting and forensic logging workflow, while full cloud storage integration is planned for future deployment. All system interactions, such as detection outputs, hash generation, and alert events, were logged to preserve forensic transparency and enable the reconstruction of evidentiary workflows.

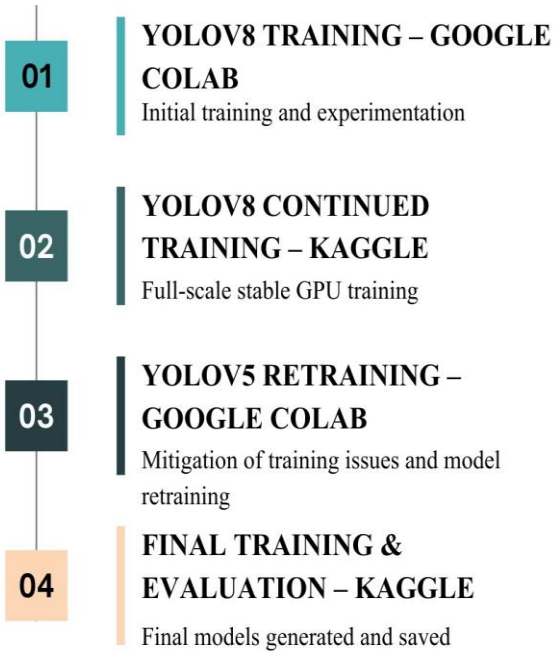

Fig. 2. VigilEye training and deployment workflow.

V. Evaluation and Limitations

A. Supporting Tools and Data Sources

These external platforms were used to assist dataset preparation, annotation, and test video acquisition, with Roboflow [33] used for dataset preprocessing, augmentation, splitting, and export, and Pexels (public video sources) used to obtain real-world scenario videos for model evaluation.

B. Evaluation Metrics and Forensic Validation

Evaluation combined traditional Computer Vision metrics with forensic-oriented validation, including stability of detections across nine datasets, error patterns under low-light and motion blur, metadata completeness, SHA-256 hash consistency, real-time inference (˜ 33.55 ms per frame), multi-threshold operational validation, and Telegram alerting effectiveness (distance ≈ 0.1402).

C. Real-Time Operational Validation

Testing on real surveillance videos demonstrated consistent inference rates across all confidence thresholds and confirmed the system's suitability for practical forensic environments.

D. Limitations and Ethical Considerations

Limitations include computational overhead since running multiple models and forensic preprocessing steps, along with environmental variability, where lighting, motion blur, and camera noise reduce detection stability. Class imbalance in certain datasets, such as blood and some accident indicators, also affected performance. Tooling constraints existed since Kaggle, Roboflow, and Google Colab all enforce feature limits and resource caps on free-tier accounts. Given that the project had no financial support, all experiments were conducted under these restricted environments, which limited training time, GPU availability, dataset size, and export options. Ethical and legal considerations included privacy risks, false positives, and surveillance accountability. Mitigation included anonymization, restricted access, and strict chain of custody management.

Although the average mAP50 result indicates promising feasibility, it is not sufficient to claim forensic-grade deployment. Future development should focus on expanding category-specific datasets, improving weaker classes such as weapons and blood, performing cross-dataset validation, and testing the system under real operational forensic conditions before practical deployment.

A further limitation is the absence of a direct quantitative baseline comparison with existing forensic surveillance systems or general-purpose detection pipelines under identical experimental settings. Future work will include controlled baseline comparisons using the same datasets, metrics, and deployment conditions.

VI. Findings

This study has shown that the VigilEye system is capable of producing consistent and dependable results for the various types of evidence recovered via surveillance through a thorough quantitative assessment, as well as numerous operational tests in real-life settings. The results of the trained models' ability to detect events included in the project indicate that several of the most critical categories of evidence, specifically, people, fire, smoke, and vehicle registration plates, yielded reliable and consistent predictions.

The results of the performance of the fire analysis dataset were highly consistent across both fire and smoke classes, and the trained model for detecting persons produced high levels of confidence and stable recognition of human subjects in a wide range of challenging environments. The model's limited detection performance in accident scenarios is due to the variety of accidents themselves. The license plate recognition model, trained on a 300-image dataset, delivered dependable performance across various lighting and distance conditions. Weapons detection showed modest to moderate performance depending on the dataset: the weapons dataset produced stronger, more isolated results. The small blood dataset consisted of 160 images. During the experiments, the lowest sensitivity for blood detection was observed as a result of the imbalanced nature of the dataset and the visual ambiguity of blood patterns, which aligns with prior discussions on visually subtle forensic classes [7]. The inability to accurately distinguish between visually similar classes was a key limiting factor; larger and more heterogeneous datasets are needed for reliable generalization in similar categories. In addition to testing the models on controlled datasets, VigilEye was tested using a mixture of real-world video footage with vehicles, pedestrians, lights, and varying light and dynamic environments. Across five different confidence thresholds, the VigilEye models maintained consistent detection abilities for all objects, including people, vehicles, fire, smoke, weapons, and license plates.

The average time spent for analysis on all datasets was about 33.55 milliseconds, indicating suitability for real-time applications. The system's effectiveness was empirically validated through rigorous forensic testing, in accordance with best practices in digital forensic video analysis. To protect privacy, facial blurring was employed to conceal the identities of detected faces, while the original, unaltered evidence was designed to be securely stored in an encrypted evidence environment, thus supporting ethical monitoring and privacy requirements.

An immediate alert was generated via Telegram upon the detection of suspicious objects and movements, enabling rapid notification to forensics in line with modern AI-based alert mechanisms. The VigilEye system maintained its consistent performance across all components of forensic evidence integrity. To ensure evidence integrity, every detection result (cropped images, metadata files, and processed frames) was hashed using the SHA-256 algorithm, in accordance with established digital tamper detection standards.

The results indicate that VigilEye can support the generation of traceable and integrity-protected forensic evidence for investigative workflows. However, formal legal admissibility would require additional validation under operational forensic procedures, institutional policies, and jurisdiction-specific legal requirements. The system shows strong performance in numerous high-risk forensic scenarios. However, improvements are still needed, particularly expanding datasets and improving models for detecting blood and other hard-to-identify forensic indicators.

As shown in Table I, the VigilEye system demonstrates a practical balance between detection accuracy and real-time performance across the evaluated forensic models.

Average performance was observed in categories with visual complexity, including people, fights (violence), and fire/smoke, while lower mAP values in weapons and blood detection are primarily attributed to dataset imbalances and visual ambiguity. Despite these variations, the system maintains effective real-time performance, with an average inference time of approximately 33.5 milliseconds per image, confirming its suitability for real-time criminal surveillance applications.

TABLE I. VIGILEYE PERFORMANCE TABLE (ALL DATASETS)

Dataset / Model	Images	Precision	Recall	mAP50	mAP50–95	Speed (ms/img)
Face Model	1000	0.9994	1.0000	0.9950	0.9509	126.85
People	690	0.7910	0.7190	0.8040	0.4040	18.15
Weapons	442	0.3150	0.3740	0.3190	0.1350	30.63
Blood	160	0.8455	0.5464	0.6688	0.3040	20.98
License Plate	300	0.8229	0.7429	0.7918	0.3968	18.94
Fight (Violence)	802	0.6884	0.6848	0.7048	0.4097	20.88
Cars	800	0.6442	0.5581	0.5884	0.3923	24.53
Car Accident	987	0.9738	0.9342	0.9523	0.9116	20.79
Fire/Smoke	400	0.7040	0.6610	0.6430	0.3440	20.23
Average (All Datasets)	–	0.7538	0.6912	0.7186	0.4720	33.5533

Fig. 3. Object detection

Fig. 3 demonstrates the performance of the VigilEye system in detecting multiple objects in a busy urban environment. Pedestrian and vehicle positions are accurately identified using surrounding boxes with associated confidence scores. This result highlights VigilEye's ability to operate effectively in complex public spaces, supporting real-time situational awareness and criminal surveillance.

Fig. 4. Fire detection

Fig. 4 shows the VigilEye system's ability to detect fire incidents along with surrounding vehicles and personnel. It assigns high confidence scores to fire zones, indicating its ability to accurately identify hazardous events in emergencies. This capability enhances the speed of incident detection and supports immediate forensic response and risk assessment.

Fig. 5. Weapon detection

Fig. 5 [25] presents the results of the VigilEye system in detecting weapons in close-range conditions, where the detected object is enclosed in a boundary square with a confidence score. This demonstrates VigilEye's effectiveness in identifying high-risk objects even under challenging visibility conditions, contributing to proactive threat identification and supporting forensic evidence.

Fig. 6. Video detection

The evaluated video footage was obtained from an opensource dataset available on Pexels [34]. The model output shown in Fig. 6 demonstrates the model's stability across five confidence thresholds (0.25, 0.20, 0.15, 0.10, and 0.05). Despite

a gradual decrease in confidence levels, the system consistently detected most key criminal elements at a crime scene, including people, vehicles, and police lights. This indicates high robustness in low confidence environments and supports the model's ability to function effectively under realistic surveillance conditions, whether visibility is poor or noise is high.

Fig. 7. License plate recognition

The license plate detection process encountered several challenges during the analysis stage. In some cases, Arabic characters were not accurately recognized, while in other instances, letters and numbers overlapped, leading to partial or incorrect detections. These issues were more noticeable when dealing with plates that contained a mix of Arabic and English characters or when the plate quality varied.

Despite these challenges, the detection system demonstrated robust performance overall, as shown in Fig. 7 [35]. The model successfully detected and recognized multiple license plates across different images, showing its capability to handle diverse plate formats.

Fig. 8. Face recognition

To protect individual privacy during surveillance and provide immediate alerts, the VigilEye system automatically applies face blurring technology before sending any external visual evidence, such as Telegram alerts. As illustrated in Fig. 8 [32], the facial region is anonymized using OpenCVbased face blurring to conceal distinctive features and protect personal privacy. Simultaneously, the original, unblurred frames are securely stored within the system and are accessible only to authorized forensic investigators, ensuring compliance with privacy requirements and proper evidence handling.

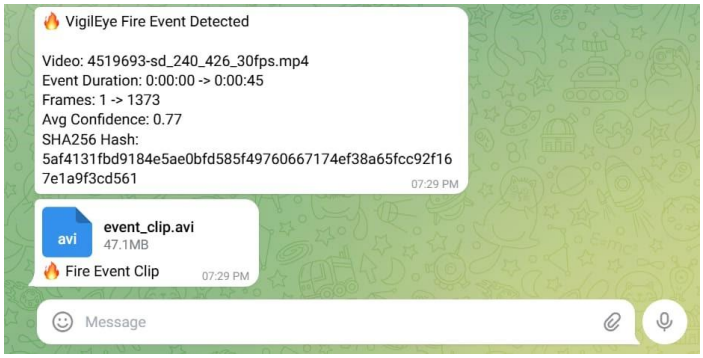

Fig. 9. Telegram alert

Fig. 9 illustrates the mechanism of instant alerts. During testing, the detection of a fire triggered an immediate notification via the Telegram app. The alert included the fire's duration, the time interval between frames, the average confidence score, and a SHA-256 hash code for evidence integrity verification. A video of the fire was also included, confirming the system's ability to detect fires in real-time, trigger alerts without delay, and preserve evidence while ensuring privacy and forensic traceability.

VII. Conclusion

Research has shown that VigilEye demonstrates the feasibility of an integrated forensic surveillance system, combining real-time monitoring with the preprocessing and secure storage of digital forensic evidence. The system demonstrated strong and consistent performance across multiple categories of forensic detection, including fires, smoke, weapons, individuals, and vehicle license plates, while maintaining operational stability under challenging environmental conditions such as low-light and poor image quality. This performance was enhanced by the integration of CLAHE-based image enhancement techniques and the safeguarding of visual forensic evidence using SHA256 hashing. Tests confirmed the system's practicality in controlled experimental settings using public datasets, simulated crime-scene scenarios, and open-source surveillance videos. However, validation using real forensic operational footage remains necessary before deployment in actual investigative environments. Integration with Telegram alerts further enhanced incident awareness, while facial blurring ensured compliance with privacy requirements by protecting biometric information without compromising original evidence. Although VigilEye achieved strong overall performance, its accuracy in detecting subtle visual evidence, such as blood and certain accident indicators, was limited by the small size and imbalance of its training datasets. These limitations highlight the need for larger and more diverse criminal datasets and for further refinement of the models for each category. Ethical and operational considerations, including privacy protection, false alarms, and transparency, remain critical in the deployment of forensic AI systems.

Overall, VigilEye demonstrates that a single, integrated platform can enhance the accuracy, integrity, and efficiency of forensic evidence. Our research helps advance digital forensic surveillance and provides a foundation for future developments, including expanded datasets, improved spatial-temporal analysis, automated chain of custody reporting, and greater interoperability with existing forensic tools.

REFERENCES

[1] J. Xiao, S. Li, and Q. Xu, "Video-based evidence analysis and extraction in digital forensic investigation," IEEE Access, vol. 7, pp. 55432–55442, 2019.

[2] V. Sonawane, R. Aaglave, R. Bedre, A. Birajdar, and V. Pardeshi, "Detection of criminal activities and anomalies through cctv's," International Research Journal on Advanced Engineering Hub (IRJAEH), vol. 10, no. 2, p. 854, 2025.

[3] A. Zubair, "Domain awareness system," Surveillance Technology Oversight Project, vol. 26, 2019.

[4] F.-L. Huang, K.-Y. Chen, and W.-H. Su, "Knowledge development trajectories of intelligent video surveillance domain: An academic study based on citation and main path analysis," Sensors, vol. 24, no. 7, p. 2240, 2024.

[5] N. Akhtar, M. Saddique, K. Asghar, U. I. Bajwa, M. Hussain, and Z. Habib, "Digital video tampering detection and localization: review, representations, challenges and algorithm," Mathematics, vol. 10, no. 2, p. 168, 2022.

[6] M. Potter, H. Gridley, N. Lichtenstein, K. Hines, J. Nguyen, and J. Walsh, "Low-light environment neural surveillance," in 2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP). IEEE, 2020, pp. 1–6.

[7] R. Pisati, R. Astya, and P. Chauhan, "A profound review of ai-driven crime detection in cctv videos," in 2024 Sixth International Conference on Computational Intelligence and Communication Technologies (CCICT). IEEE, 2024, pp. 193–199.

[8] N. Ahmed, M. E. Hossain, Z. Hossain, M. F. Kabir, and I. S. Hossain, "Assessing the potential and ethical implications of agentic ai in surveillance technology," Formosa Journal of Multidisciplinary Research, vol. 4, no. 4, pp. 1841–1858, 2025.

[9] I. A. Shah, N. Jhanjhi, and R. M. A. Ujjan, "An intelligent and efficient approach for a weapon detection system using computer vision and edge computing," Engineering Proceedings, vol. 82, no. 1, p. 117, 2024.

[10] A. Dutta, P. Boral, and G. Suseela, "Intelligent image sensing for crime analysis: A ml approach towards enhanced violence detection and investigation," arXiv preprint arXiv:2506.13910, 2025.

[11] P. V. B. Ngoc, N. H. Nguyen, N. L. Thien, V. T. Doan et al., "Realtime fire and smoke detection for trajectory planning and navigation of a mobile robot," Engineering, Technology & Applied Science Research, vol. 13, no. 5, pp. 11843–11849, 2023.

[12] A. Khalil, F. Alam, D. Shah, I. Khalil, S. Ali, and M. Tahir, "Real time blood detection in cctv surveillance using attention enhanced inceptionv3," Scientific Reports, vol. 15, no. 1, p. 28977, 2025.

[13] A. Hamdi, Y. K. Chan, and V. C. Koo, "A new image enhancement and super resolution technique for license plate recognition," Heliyon, vol. 7, no. 11, 2021.

[14] T. Vats, D. Chakraborty, D. Rudrapal, and B. Bhattacharya, "A hybrid yolo-vit approach for real-time accident monitoring through automated cctv surveillance," in 2025 IEEE Guwahati Subsection Conference (GCON). IEEE, 2025, pp. 1–6.

[15] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lucic˘, and C. Schmid,´ "Vivit: A video vision transformer," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 6836–6846.

[16] M. R. Sholahuddin, M. Harika, I. Awaludin, Y. C. Dewi, F. D. Fauzan, B. P. Sudimulya, and V. P. Widarta, "Optimizing yolov8 for real-time cctv surveillance: A trade-off between speed and accuracy," Jurnal Online Informatika, vol. 8, no. 2, pp. 261–270, 2023.

[17] D. Wang, Z. Gao, J. Fang, Y. Li, and Z. Xu, "Eagle-yolov8: Uav object detection inspired by the eagle-eye vision system," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2025.

[18] M. Kassir, S. Haidar, and A. Yaacoub, "Exploring personalized federated learning architectures for violence detection in surveillance videos," arXiv preprint arXiv:2504.00857, 2025.

[19] Diksha, Aman, and A. Chadha, "Review paper on face recognition attendance system using python," International Journal of Advances in Engineering and Management (IJAEM), vol. 6, no. 5, pp. 775–781, 2024.

[20] F. P. Mahdi, M. M. Habib, M. A. R. Ahad, S. Mckeever, A. Moslehuddin, and P. Vasant, "Face recognition-based real-time system for surveillance," Intelligent Decision Technologies, vol. 11, no. 1, pp. 79–92, 2017.

[21] T. Mukhiddin, H. R. Arousha, A. Ubaydullo, L. Wookey, and S. Lee, "Privacy-preserving of human identification in cctv data using a novel deep learning-based method," in 2022 IEEE International Conference on Big Data and Smart Computing (BigComp). IEEE, 2022, pp. 211–214.

[22] Scientific Working Group on Digital Evidence (SWGDE), "Swgde best practices for digital evidence collection and preservation," Online, 2023, available online: https://swgde.org/documents. Accessed: May 15, 2026.

[23] K. Kent, S. Chevalier, and T. Grance, "Guide to integrating forensic techniques into incident response," NIST Special Publication 800-86, 2006.

[24] R. Universe, "Fire detection dataset for vigileye," Roboflow Universe, 2025, available online: https://universe.roboflow.com/vigileye/ fire-h2gkf-zrl1m. Accessed: May 15, 2026.

[25] Roboflow Universe, "Weapons dataset," Roboflow Universe, 2025. [Online]. Available: https://universe.roboflow.com/vigileye-v0y1v/weapons-rvq2k-pgoem. Accessed: May 15, 2026.

[26] Roboflow Universe, "People detection dataset," Roboflow Universe, 2023, archived dataset, previously hosted on Roboflow Universe. Accessed: May 15, 2026.

[27] Roboflow Universe, "Blood detection dataset," Roboflow Universe, 2025. [Online]. Available: https://app.roboflow.com/vigileye/blood-3fivf-titb8/1. Accessed: May 15, 2026.

[28] Roboflow Universe, "Car accident dataset," Roboflow Universe, 2025. [Online]. Available: https://app.roboflow.com/vigileye/car-accident-imv7t-pbjii/1. Accessed: May 15, 2026.

[29] Roboflow Universe, "License plate recognition dataset," Roboflow Universe, 2025. [Online]. Available: https://app.roboflow.com/vigileye/license-plate-recognition-quhlg-ub2f0. Accessed: May 15, 2026.

[30] Roboflow Universe, "Fight detection computer vision dataset," Roboflow Universe, 2025. [Online]. Available: https://universe.roboflow.com/custom-yolov5-dx8he/fight-lpfb6. Accessed: May 15, 2026.

[31] Roboflow Universe, "Cars computer vision dataset," Roboflow Universe, 2025. [Online]. Available: https://universe.roboflow.com/object-detection-training-54jsr/cars-zwxpr. Accessed: May 15, 2026.

[32] Roboflow Universe, "Face detection dataset," Roboflow Universe, 2025. [Online]. Available: https://app.roboflow.com/vigileye/face-detection-wetml-v6eq4. Accessed: May 15, 2026.

[33] J. Nelson, B. Dwyer, and J. Solawetz, "Roboflow: Computer vision tools and datasets," Roboflow, 2022, available online: https://roboflow.com. Accessed: May 15, 2026.

[34] K, "Parked police cars," Pexels, 2020, available online: https:// www.pexels.com/video/parked-police-cars-6581006/. Accessed: May 15, 2026.

[35] R. Universe, "License plate recognition model for vigileye," Roboflow, 2025, available online: https://app.roboflow.com/vigileye/ license-plate-recognition-quhlg-ub2f0/models. Accessed: May 15, 2026.